End-to-End Data Transfer Systems for 100 Gbps Networks and Beyond

Yufei Ren *Stony Brook University*

Advisors: Dantong Yu *Brookhaven National Laboratory*

Shudong Jin *Stony Brook University*

Stony Brook University

BROOKHAVEN NATIONAL LABORATORY

Introduction

Data-intensive applications such as those in grid and cloud computing environment are generating extremely high volumes of data. Data is often transferred, visualized, and analyzed by geographically distributed teams of users. Although high performance network capabilities, such as 40/56/100 Gbps, are (becoming) available to support these applications across both local and wide area networks, existing TCP-based data transfer applications (GridFTP, bbcp, and rsync) and storage area networks (iSCSI) cannot fully utilize the benefit of state-of-the-art hardware such as remote direct memory access (RDMA) and non-uniform memory access (NUMA).

The efficient design of network protocols and software systems is a crucial aspect of research and development in data-intensive computing. My research focuses on designing and implementing an end-to-end data transfer system for 100 Gbps networks and beyond. The whole system is divided into a front-end data transfer part and a back-end data storage one, as shown in Figure 1.

- **Front-end: Design and implement an RDMA-based parallel data transfer protocol and an RDMA-enabled FTP software: RFTP.**
- **Back-end: Design a NUMA-aware cache for iSCSI protocol, and implement the cache within Linux SCSI Target Framework ,and its iSCSI and iSER drivers.**

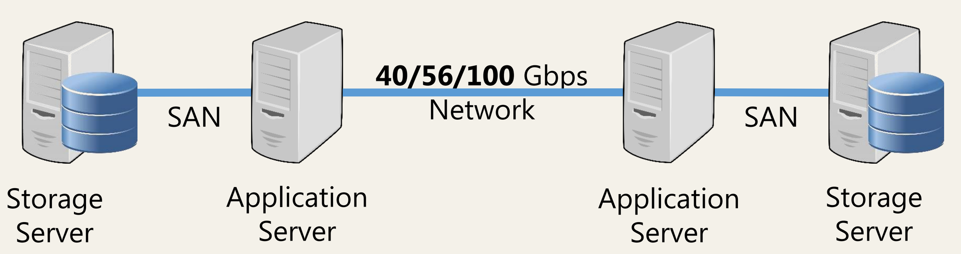

Figure 1: End-to-End Data Transfer System.

RDMA-Based Data Transfer Protocol and Software

In the front-end, my research studies the design and performance issues of data transfer tools for high-speed networks such as state-of-the-art 40 Gbps Ethernet and 56 Gbps InfiniBand. RDMA offers zero-copy and kernel bypass mechanism and achieves high throughput with low processing overhead. In the meantime, RDMA also introduces programming complexities including explicit credit management, explicit memory management, and asynchronous and event-driven programming interfaces.

- **RDMA Pros**
  - Zero-copy, kernel bypass
  - Higher throughput and lower latency
- **RDMA Cons**
  - Sophisticated credit-based message exchange
  - Difficult memory management
  - Asynchronous and event-driven programming

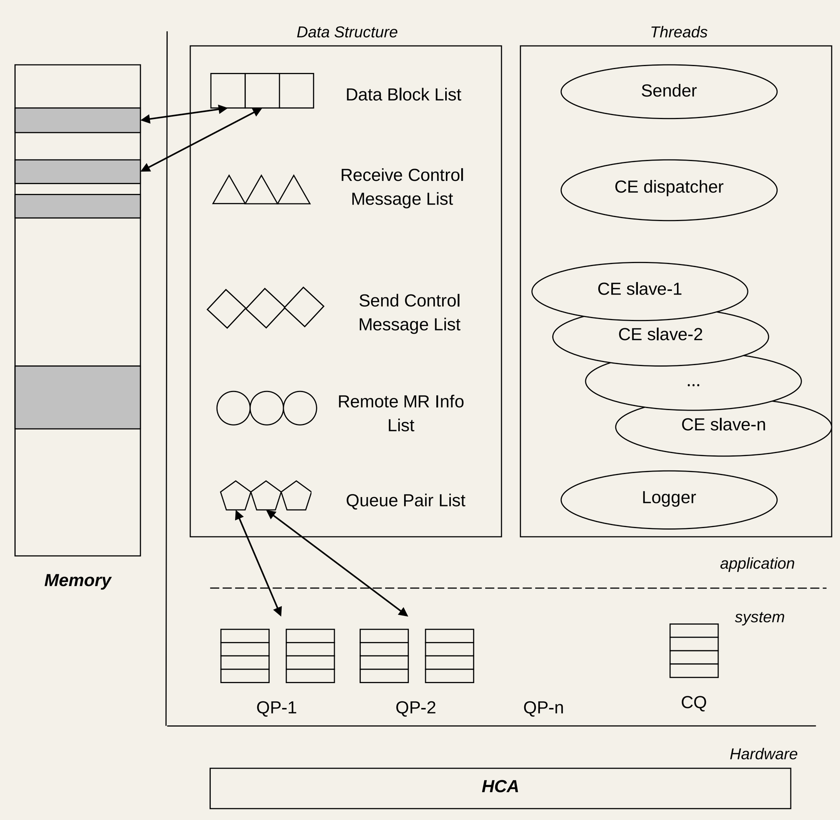

Figure 2: Multi-threaded architecture and data structure of RDMA-based middleware.

By taking consideration both advances and drawbacks of RDMA technology, my research designs a middleware that can take advantage of RDMA techniques to attain high network throughput. It provides the necessary data communication and access functions, while maximizing the parallelism of data processing with advanced features such as zero-copy, reuse of memory regions, multi-stream parallel transfer, and multi-threading, as shown in Figure 2.

Based on this middleware, my research develops an RDMA-based FTP service: RFTP. It reserved the framework of FTP protocol (RFC 959), defined a series of new commands for RDMA features including RADR, RSTR, and RRTR, and implemented RDMA extensions in Linux FTP service.

NUMA-Aware Cache for iSCSI Storage

In the back-end, my research leverages the iSCSI protocol to construct high performance storage area networks. During testing iSCSI in NUMA environment, we found that the existing iSCSI target software often dispatches an access request with cache hit to an I/O thread that is not local the cached data, and thus cannot fully utilize the new multi-core power. Because NUMA is widely used to increase the computation density per host in data centers, we design a NUMA-aware cache mechanism to align cache memory with the local NUMA node and schedule I/O requests to those threads that are local to the data to be accessed. This NUMA-aware solution can result in a lower access latency and higher system throughput.

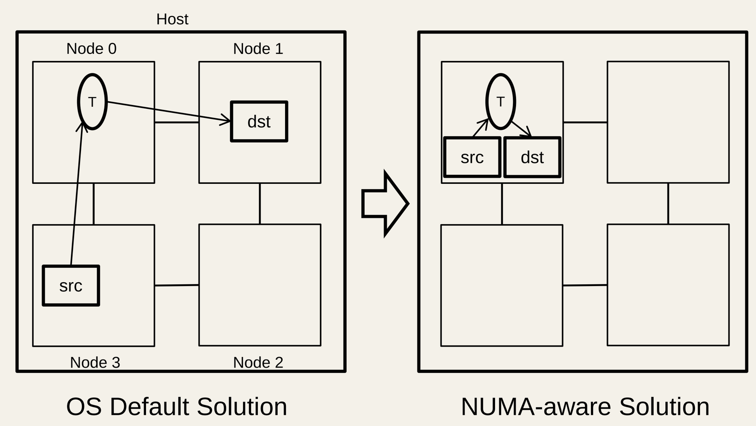

Figure 3: Memory copy routine on a four-node NUMA host.

My research designs and implements a NUMA-aware cache in Linux SCSI Target Framework. As depicted in Figure 4, each NUMA node contains its own NUMA-aware cache, network buffers, and worker thread groups. Different from the existing framework, the worker threads in this design try to copy data from their local cache to a local network buffer on a cache-hit.

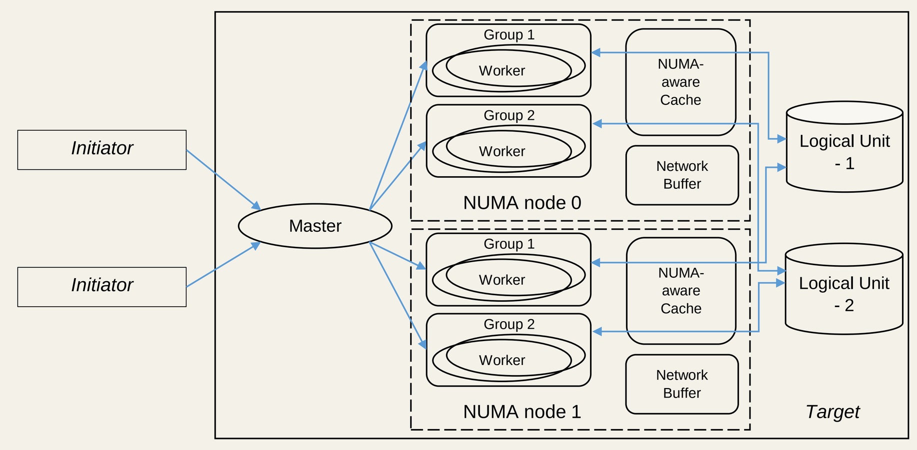

Figure 4: Constructing a NUMA-aware cache for iSCSI target.

Data Transfer Experimental Results

End-to-end experimental results in high-speed LAN.

- **Front-end: 3 x 40 Gbps RoCE**
- **Back-end: 2 x 56 Gbps InfiniBand**

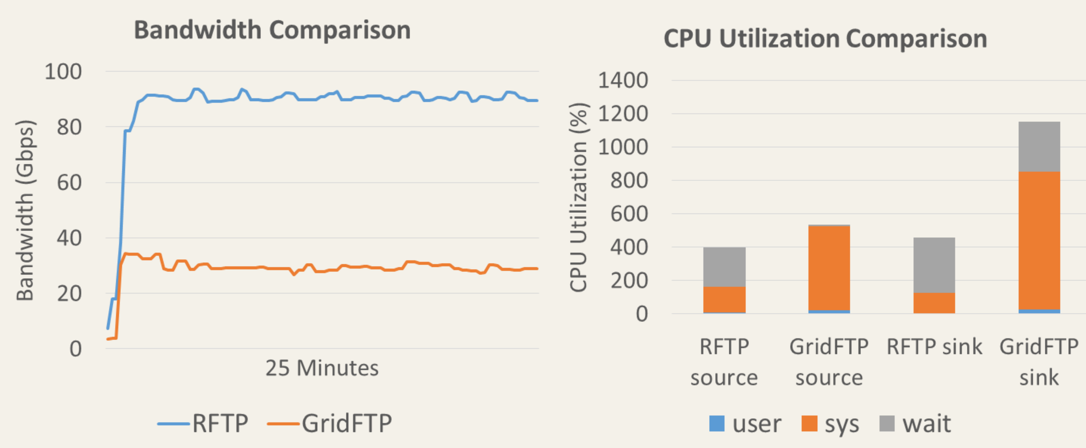

Figure 5: Bandwidth and CPU utilization comparison between RFTP and GridFTP in LAN.

Experimental results over 40 Gbps WAN.

- **40 Gbps RoCE WAN**
- **4,000 miles**
- **Loopback from NERSC to ANL, and back to NERSC**
- **RTT: 95 millisecond**
- **BDP: 500 MB**

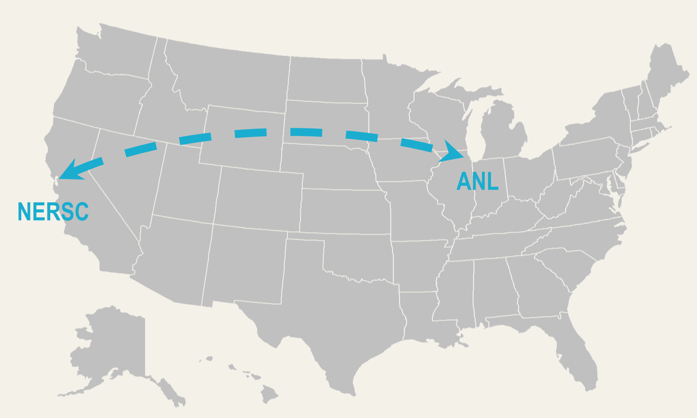

Figure 6: WAN link between NERSC and ANL.

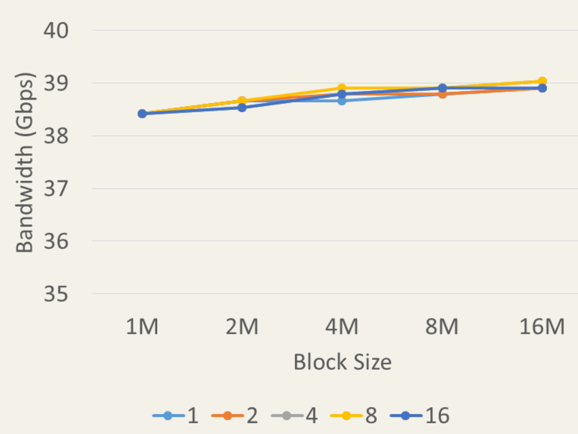

Figure 7: RFTP bandwidth in WAN.

NUMA-Aware Cache Experimental Results

Experimental results on a 4-node NUMA system.

- **Target: Dell R820 - 4-node NUMA system**
- **Initiators: 2 IBM X3650 and 2 HP DL 380**
- **Network: 2 x 56 Gbps InfiniBand on each host**

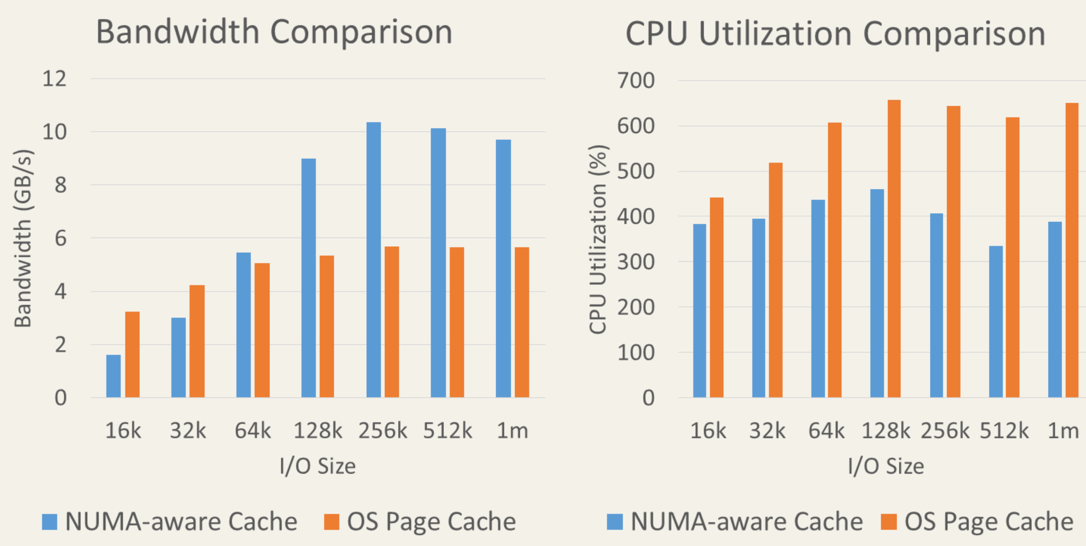

Figure 8: Bandwidth and CPU utilization comparison NUMA-aware cache and OS page cache.
Note: there are 2 concurrent I/O threads in each initiator.

Future Work

My research continuous on designing flexible, auto-tuned end-to-end data transfer solutions, developing new system software, and evaluating them in real networks and systems. Future work includes:

- **More considerations on packet loss environment.**
- **Efficient transfer for large amount of small files.**
- **Evaluate NUMA-aware cache on large scale NUMA hosts.**
- **High efficient, lockless memory management on NUMA system.**

Selected Publication

Design and Performance Evaluation of NUMA-Aware RDMA-Based End-to-End Data Transfer Systems
Yufei Ren, Tan Li, Dantong Yu, Shudong Jin, Thomas Robertazzi
Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis **SC '13**, Denver, Colorado, November 2013.

Characterization of Input/Output Bandwidth Performance Models in NUMA Architecture for Data Intensive Applications
Tan Li, **Yufei Ren**, Dantong Yu, Shudong Jin, Thomas Robertazzi
Proceedings of the International Conference on Parallel Processing **ICPP '13**, Lyon, France, October 2013.

Design and Testbed Evaluation of RDMA-Based Middleware for High-Performance Data Transfer Applications
Yufei Ren, Tan Li, Dantong Yu, Shudong Jin, Thomas Robertazzi
***Journal of Systems and Software**, Volume 86, Issue 7, July 2013, Pages 1850-1863, 10.1016/j.jss.2013.01.070.*

Protocols for Wide-Area Data-Intensive Applications: Design and Performance Issues
Yufei Ren, Tan Li, Dantong Yu, Shudong Jin, Thomas Robertazzi, Brian L. Tierney, Eric Pouyoul
Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis **SC'12**, Salt Lake City, Utah, November 2012.

Middleware Support for RDMA-Based Data Transfer in Cloud Computing
Yufei Ren, Tan Li, Dantong Yu, Shudong Jin, Thomas Robertazzi
Proceedings of the High-Performance Grid and Cloud Computing Workshop (held in conjunction with IPDPS 2012), Shanghai, China, May 2012.

Software

RFTP: http://ftp100.cewit.stonybrook.edu/rftp

Acknowledgements

This research is grateful to the facility and hardware donation from Mellanox Technologies, Inc., HotLava Systems, Chelsio Communications, LSI Corp., and Fusion-io, Inc. This research is supported by United States Department of Energy, Grant No. DE-SC0003361.